Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Cancelled).

2. (Previously presented) A video file server for providing clients with video-on-demand access to movies, the video file server comprising:

a cached disk storage system including a primary cache and disk storage for storing the movies; and

a multiplicity of data mover computers coupled to the cached disk storage system for streaming video data from the cached disk storage system to clients in a data network, each of the data mover computers having a local cache;

wherein the movies are ranked with respect to popularity, and a respective set of the data movers are pre-assigned for servicing video streams for each movie ranking; and

wherein the data movers in the respective sets of data movers are configured differently for providing more network interface resources for very popular movies and for providing more local cache memory resources for less popular movies.

3. (Original) The video file server as claimed in claim 2, wherein for very popular movies, the very popular movies are retained in their entirety in local cache of the data movers assigned to service the very popular movies.

4. (Previously presented) The video file server as claimed in claim 2, wherein the sets of data movers include a set consisting of more than one data mover for servicing one very popular movie, a set consisting of one data mover for servicing only one movie, and a set consisting of one data mover for servicing a plurality of the movies.

5. (Previously presented) The video file server as claimed in claim 2, wherein a series of at least some of the data movers include direct links for transfer of movie data from a data mover set servicing one movie ranking to a data mover set servicing a next higher movie ranking and for transfer of movie data from the data mover set servicing the one movie ranking to a data mover set servicing a next lower movie ranking.

6. (Previously presented) The video file server as claimed in claim 2, wherein data mover resources for a certain number of video streams from the data movers to the clients are reserved for each of a multiplicity of the movies.

7. (Previously presented) The video file server as claimed in claim 2, wherein the video file server is programmed for locking in the primary cache a plurality of entire movies, and when there is a need for servicing a more popular movie from the primary cache and there is insufficient free cache memory for servicing the more popular movie from the primary cache, transferring the servicing of a less popular movie from the primary cache to disk storage in order to free cache memory for servicing the more popular movie from the primary cache.

8. (Original) The video file server as claimed in claim 7, wherein the video file server is programmed for freeing primary cache memory by transferring the servicing of a least popular movie in the primary cache from the primary cache to the disk storage so long as no more than a certain number of video streams are being serviced concurrently from the least popular movie in the primary cache.

9. (Previously presented) The video file server as claimed in claim 2, wherein the video file server is programmed for negotiating with a client for selection of an available movie during peak demand when resources are not available to select freely any movie in the disk storage for which a video stream can be started.

10. (Cancelled).

11. (Currently Amended) The video file server as claimed in claim [[10]] 12, wherein the video file server is programmed for freeing locked cache memory by transferring the servicing of the least popular movie in the cache from the cache to the disk storage so long as no more than a certain number of video streams are being concurrently serviced from the least popular movie in the cache.

12. (Currently Amended) ~~The video file server as claimed in claim 10,~~
A video file server for providing clients with video-on-demand access to movies, the video file server comprising:

a cached disk storage system including a cache and disk storage for storing the movies; and

a multiplicity of data mover computers coupled to the cached disk storage system for streaming video data from the cached disk storage system to clients in a data network;

wherein the video file server is programmed for locking in the cache a plurality of entire movies, and when there is a need for servicing a more popular movie from the cache and there is insufficient free cache memory for servicing the more popular movie from the cache, transferring the servicing of a less popular movie from the cache to disk storage in order to free cache memory for servicing the more popular movie from the cache.

wherein each of the data mover computers has a local cache, the movies are ranked with respect to popularity, and a respective set of the data movers are pre-assigned for servicing video streams for each movie ranking, and the data movers in the respective sets of data movers are configured differently for providing more network interface resources for very popular movies and for providing more local cache memory resources for less popular movies.

13. (Currently amended) The video file server as claimed in claim [[10]] 12, wherein a series of at least some of the data movers include direct dedicated links for transfer of movie data from a data mover set servicing one movie ranking to a data mover set servicing a next higher movie ranking and for transfer of movie data from the data mover set servicing the one movie ranking to the data mover set servicing a next lower movie ranking.

14. (Currently amended) The video file server as claimed in claim [[10]] 12, wherein data mover resources for a certain number of video streams from the data movers to the clients are reserved for each of a multiplicity of the movies.

15. (Cancelled).

16. (Previously presented) A method of operating a video file server for providing clients with video-on-demand access to movies, the video file server having a cached disk storage system including a primary cache and disk storage containing the movies, and a multiplicity of data mover computers coupled to the cached disk storage system for streaming video data from

the cached disk storage system to clients in a data network, each of the data mover computers having a local cache, wherein the method includes:

ranking the movies with respect to popularity, and assigning a respective set of the data movers to each movie ranking, and

servicing video streams for each movie ranking with the respective set of data movers assigned for servicing said video streams for said each movie ranking; and

which includes configuring differently the data movers in the respective sets of data movers in order to provide more network interface resources for very popular movies and for providing more local cache memory resources for less popular movies.

17. (Previously presented) The method as claimed in claim 16, which includes, for very popular movies, retaining the very popular movies in their entirety in the local cache of the data movers assigned to service the very popular movies.

18. (Previously presented) The method as claimed in claim 16, which includes servicing a most popular movie with an assigned data mover set consisting of more than one data mover, servicing only one movie with an assigned data mover set consisting of one data mover, and servicing a plurality of movies with an assigned data mover set consisting of one data mover.

19. (Previously presented) The method as claimed in claim 16, wherein a series of at least some of the data movers are linked by direct dedicated data links and the method includes

transferring movie data from a data mover set servicing one movie ranking to a data mover set servicing a next higher movie ranking and transferring movie data from a data mover set servicing the one movie ranking to a data mover set servicing a next lower movie ranking.

20. (Previously presented) The method as claimed in claim 16, which includes reserving data mover resources for a respective number of video streams from the data movers to the clients for each of a multiplicity of the movies.

21. (Previously presented) The method as claimed in claim 16, which includes locking in the primary cache a plurality of entire movies, and when there is a need for servicing a more popular movie from the primary cache and there is insufficient free cache memory for servicing the more popular movie from the primary cache, transferring the servicing of a less popular movie from the primary cache to the disk storage in order to free primary cache memory for servicing the more popular movie from the primary cache.

22. (Previously presented) The method as claimed in claim 16, which includes freeing primary cache memory by transferring the servicing of a least popular movie in the primary cache from the primary cache to the disk storage so long as no more than a certain number of video streams are being concurrently serviced from the least popular movie in the primary cache.

23. (Previously presented) The method as claimed in claim 16, which includes the video file server negotiating with a client for selection of an available movie during peak demand when

resources are not available to select freely any movie in the disk storage for which a video stream can be started.

24. (Cancelled).

25. (Currently amended) The method as claimed in claim [[24]] 26, which includes the video file server freeing locked cache memory by transferring the servicing of a least popular movie in the cache from the cache to the disk storage so long as no more than a certain number of video streams are being concurrently serviced from the least popular movie in the cache.

26. (Currently amended) ~~The method as claimed in claim 24,~~
A method of operating a video file server for providing clients with video-on-demand access to movies, the video file server having a cached disk storage system including a cache and disk storage containing the movies, and a multiplicity of data mover computers coupled to the cached disk storage system for streaming video data from the cached disk storage system to clients in a client data network, the method comprising:

locking in the cache a plurality of entire movies, and when there is a need for servicing a more popular movie from the cache and there is insufficient free cache memory for servicing the more popular movie from the cache, transferring the servicing of a less popular movie from the cache to the disk storage in order to free cache memory for servicing the more popular movie from the cache,

wherein each of the data mover computers has a local cache, the method includes ranking the movies with respect to popularity, assigning a respective set of the data movers for servicing video streams for each movie ranking, and configuring the data movers in the respective sets of data movers differently for providing more network interface resources for very popular movies and for providing more local cache memory resources for less popular movies.

27. (New) The video file server as claimed in claim 2, wherein the respective sets of data movers are configured differently by having fewer cache memory resources and more network interface resources in the data movers that service more popular movies than in the data movers that service less popular movies.

28. (New) The video file server as claimed in claim 12, wherein the respective sets of data movers are configured differently by having fewer cache memory resources and more network interface resources in the data movers that service more popular movies than in the data movers that service less popular movies.

29. (New) The method as claimed in claim 16, wherein the respective sets of data movers are configured differently by providing fewer cache memory resources and more network interface resources in the data movers that service more popular movies than in the data movers that service less popular movies.

30. (New) The method as claimed in claim 26, wherein the respective sets of data movers are configured differently by providing fewer cache memory resources and more network interface resources in the data movers that service more popular movies than in the data movers that service less popular movies.